「システムLSI検証」考

第３回　プロセスを見直そう

渋谷康洋

本コラムの第2回（本誌2005年12月号，p.131）では，「オブジェクト指向」の推薦書をいくつか紹介しました．読んでいただけましたか．読まれた方は，どのような感想を持たれたでしょうか？　「言いたいことはわかった．モデル化とfile/classはスッキリ整理できた．でも，そこからもう1歩進んだ，LSI検証に直接役立つ教科書はないの？」．残念ながら，まだないようです．

● 検証に至るまでの手順（プロセス）は正しいか？

要求仕様書から始まり，RTLコーディングを行い，そしてシミュレーションに至るまでのみなさんの作業手順はどのようなものでしょうか．**図1**を見ながら，簡単におさらいしてみましょう．

まず，要求仕様書が提示されます．その要求仕様書をもとに，適切なブロック分割を行い，ざっくりとした性能検討やリソース共有の検討，そして詳細な設計へと進みます．設計仕様書にはタイミング・チャートやレジスタ，モジュール・ピン情報などをまとめ，コーディングのときに迷いが生じないレベルにしあげます．そして，社内の設計ルールなどに準じてコーディングを行います．コーディングのあとはLintツールなどを通して，いよいよシミュレーションです．おおむね，このような手順を踏むことが多いかと思います．

さて，最近ではこのシミュレーションという段階が様変わりして，以前のようにハードウェア技術者がtaskを後からガチガチと書き並べることは少なくなってきました．検証専任のエンジニアが張り付き，RTLコードがあがってくるまでに，検証環境を整えて提供する体制を採るところも増えています．ただし，実情はというと，「どうにか，がんばって」間に合わせている，といった感じです．

なぜでしょう？　なぜ，そんなに無理な状況になっているのでしょう？　検証エンジニアが足りないから？

確かに検証エンジニアは不足しています．しかし，もっと基本的な部分に問題があるように思います．多くの場合，検証エンジニアが呼ばれるのは，シミュレーションの直前か，あるいはすでに部分的なシミュレーションが始まったあたりからです．つまり，どういうわけか，すでに話が盛り上がってからお声がかかります．これでは，タイミングが遅すぎます．

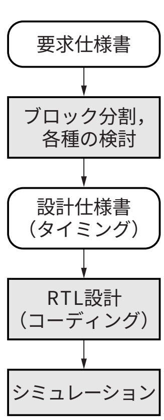


図1
LSI設計フロー

● なぜそうなるのか，状況を分析してみよう

「タイミングが遅い」と言うだけでは解決に至らないので，なぜそうなるのか，分析してみましょう．

わりに多いと思われるのが，設計者がRTLコーディングのあとで検証環境の作成に入るため，自然に検証環境の構築が後工程となってしまうケースです．これはある意味しかたがないのかもしれません．別に，無理に検証エンジニアに並行開発を依頼しなくても，「手順が正しければ」設計者が正しい検証環境を作成することはできます．

ここで，「手順が正しければ」という点が問題です．そうです．検証環境や検証モデルを作成するとき，検証設計図や検証仕様書の作成といった手順が欠落していませんか？　RTL設計ではブロックごと，あるいは全体の設計仕様書をまとめているのに，検証環境の作成ではなぜそういった手順を踏めないのでしょうか？

もしかすると，「アノテートは要らない．ACスペックもないし，フォルス・パス指定もない」など，伝統的なハードウェア技術者の経験則や心理が悪い方向に働き，意識が甘くなっているのかもしれません．だとしたら，こういった悪い認識はあらためましょう．ちなみに筆者も以前はそういうところがありました．反省．

別の背景として，「検証設計図の作成ガイドライン」のようなものがこれまでなかったので，これを作成してこなかったのかもしれません．しかし，だからといって，設計図や仕様書なしで開発を始めていいわけはありません．少なくともRTL設計ではやりませんよね．ブロック分割もタイミングの算段もピン情報もなく，いきおいでコードを書き始めるなんて…．

検証環境や検証モデルの作成も同じです．大事な，そしてたいへん難しいしごとだからこそ，踏むべき手順をおろそかにしてはいけないのです．

＜筆者プロフィール＞
渋谷康洋．1987年，タイトー入社，中央研究所に配属．業務用ゲーム・ハードウェアを支えた10年．1996年，リアルビジョン勤務．3D-ASICを支えた3年．Palo Altoの3Dベンチャの解散を受けて，2000年からLSIシステムズに勤務．受託設計やSpecmanEliteを5年間支え，現在，日本ケイデンス・デザイン・システムズ社に勤務．EDA屋としてみなさんを支える立場に．